Fig. 1

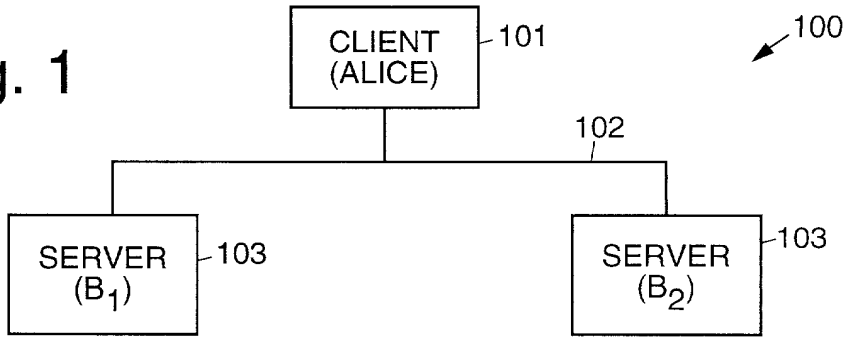
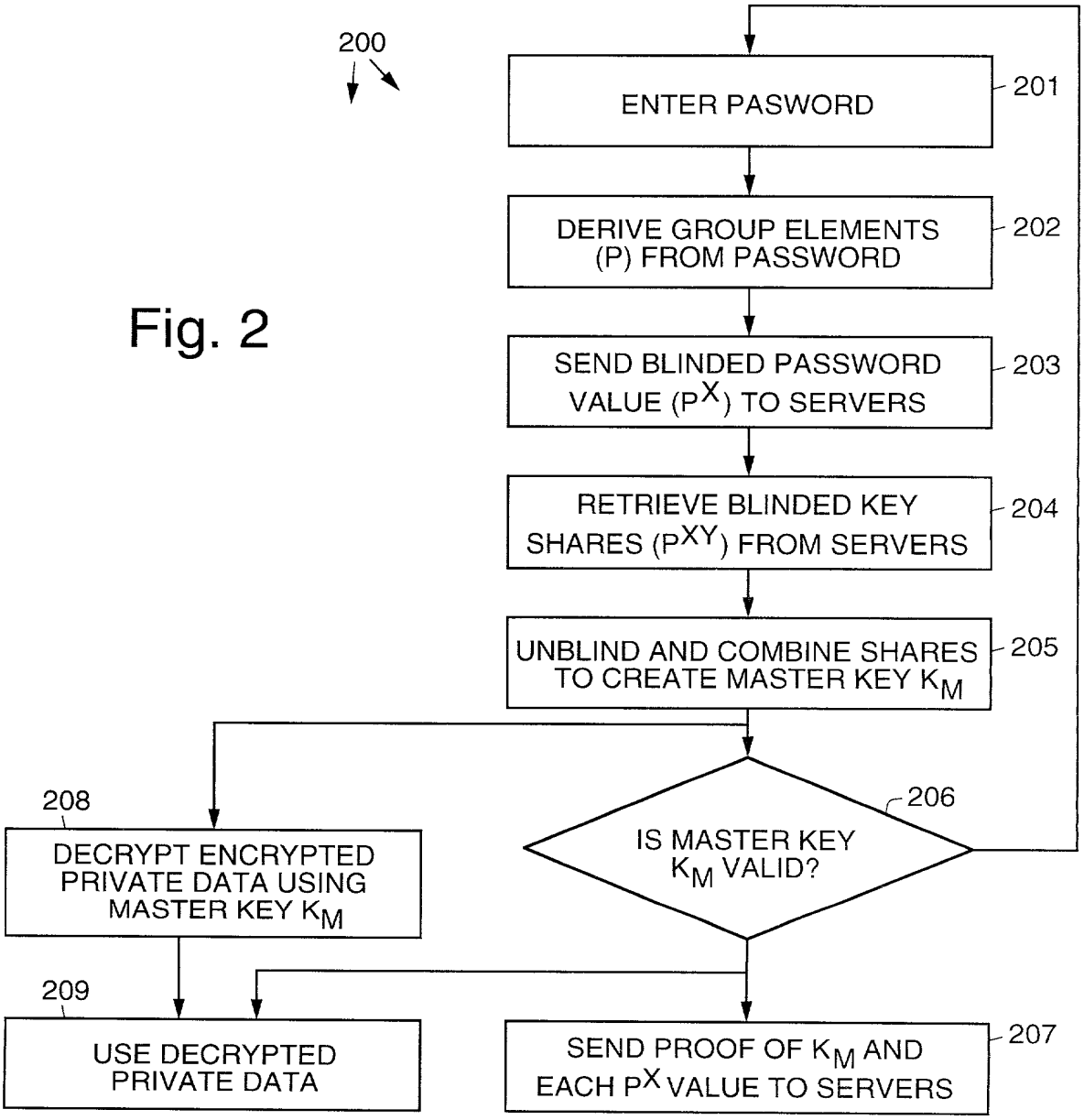

CLIENT
(ALICE)
101
100
102
SERVER
(B_1)
103
SERVER
(B_2)
103

Fig. 2


200
ENTER PASWORD
201
DERIVE GROUP ELEMENTS
(P) FROM PASSWORD
202
SEND BLINDED PASSWORD
VALUE (P^X) TO SERVERS
203
RETRIEVE BLINDED KEY
SHARES (P^{XY}) FROM SERVERS
204
UNBLIND AND COMBINE SHARES
TO CREATE MASTER KEY K_M
205
IS MASTER KEY
K_M VALID?
206
208
DECRYPT ENCRYPTED
PRIVATE DATA USING
MASTER KEY K_M
209
USE DECRYPTED
PRIVATE DATA
SEND PROOF OF K_M AND
EACH P^X VALUE TO SERVERS
207

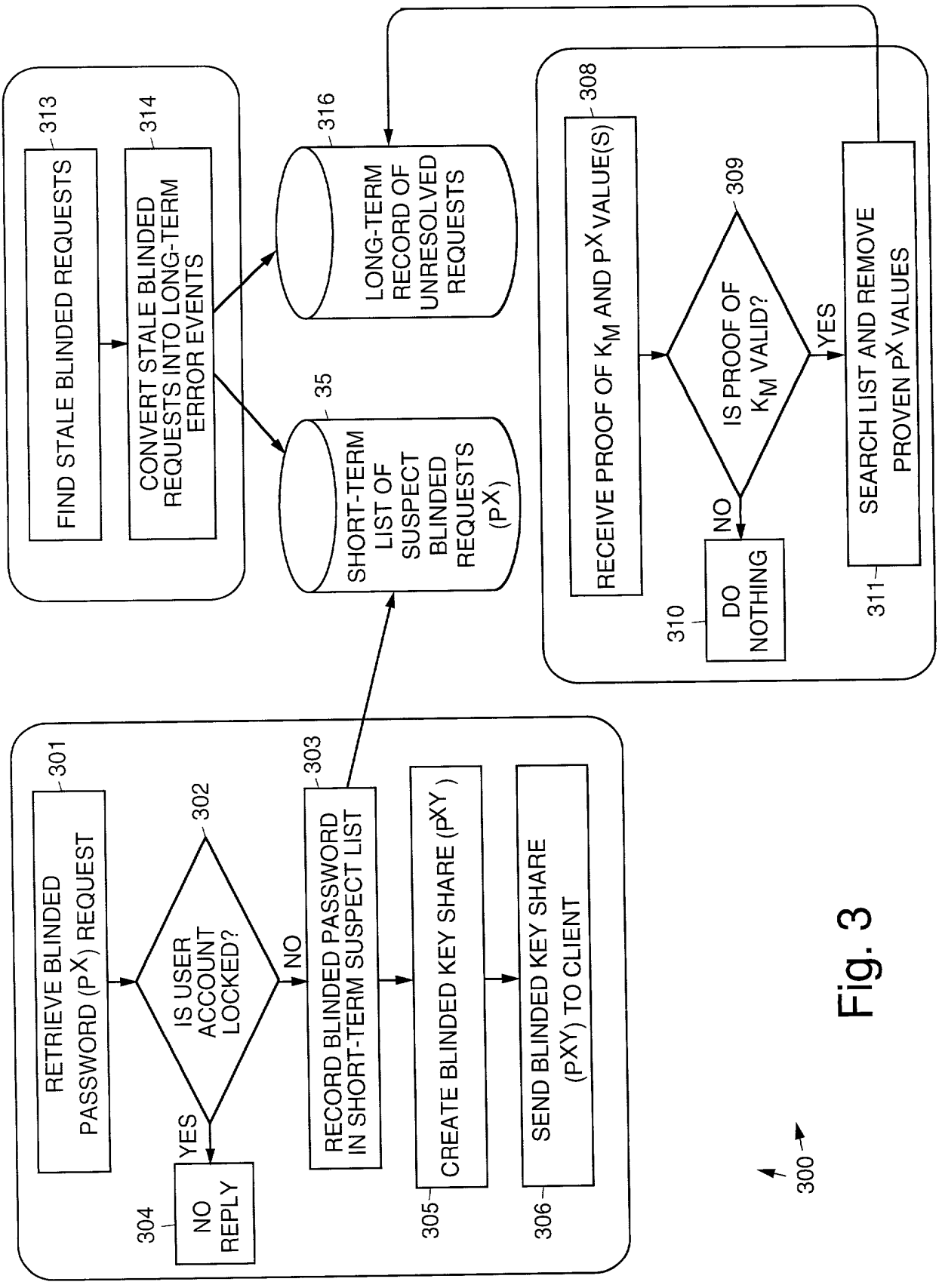

RETRIEVE BLINDED PASSWORD (P^X) REQUEST
301
IS USER ACCOUNT LOCKED?
302
YES
NO
NO REPLY
304
RECORD BLINDED PASSWORD IN SHORT-TERM SUSPECT LIST
303
CREATE BLINDED KEY SHARE (P^{XY})
305
SEND BLINDED KEY SHARE (P^{XY}) TO CLIENT
306
SHORT-TERM LIST OF SUSPECT BLINDED REQUESTS (P^X)
35
LONG-TERM RECORD OF UNRESOLVED REQUESTS
316
FIND STALE BLINDED REQUESTS
313
CONVERT STALE BLINDED REQUESTS INTO LONG-TERM ERROR EVENTS
314
RECEIVE PROOF OF K_M AND P^X VALUE(S)
308
IS PROOF OF K_M VALID?
309
NO
YES
DO NOTHING
310
SEARCH LIST AND REMOVE PROVEN P^X VALUES
311
300

Fig. 3

Fig. 4

400

	Alice	B_1	B_2	Directory
401		{UserID, y_1, V}	{UserID, y_2, V}	{UserID, *encrypt*(K_m{H_P, U})}
402	P = *func*(password)			
403	$m_1 = P^x$			
404	UserID,$m_1 \rightarrow B_1, B_2$			
405		$m_2 = m_1{}^{y1}$		
407		record m_2		
407		Alice $\leftarrow m_2$		
408				{Alice $\leftarrow$ *encrypt*(K_m{H_P, U})}
409			$m_3 = m_1{}^{y2}$	
410			record m_1	
411			Alice $\leftarrow m_3$	
412	K_m= *hash*(m_2 * m_3)$^{1/x}$ = *hash*(K_1 * K_2)			
413	decrypt encrrypted data using K_m to get H_P and U			
414 415	if H_P != hash(P), abort			
416	m_4 = *sign*(U, m_1) $\rightarrow B_1, B_2$			
417		verify m_4 signature of m_1 using V	verify m_4 signature of m_1 using V	
418 419		if verified, erase m_1 event	if verified, erase m_1 event	

	Alice	B_1	B_2	Directory
501		{UserID, y_1, V}	{UserID, y_2, V}	{UserID, *encrypt*(K_m{H_P, U})}
502	P = *func*(password)			
503	$m_1 = P^x$			
504	UserID,$m_1 \rightarrow B_1$			
505		$m_2 = m_1^{y_1}$		
506		record m_1		
507		UserID, $m_2 \rightarrow B_2$		
508				{Alice $\leftarrow$ *encrypt*(K_m{H_P, U})}
509			$m_3 = m_2^{y_2}$	
510			record m_3	
511		Alice $\leftarrow m_3$	$B_1 \leftarrow m_3$	
512	K_m= *hash*($m_3^{1/x}$) = *hash*($P^{y_1 y_2}$)			
513	decrypt encrrypted data using K_m to get H_P and U			
514 515	if H_P != hash(P), abort			
516	m_4 = *sign*(U, m_1, m_3) $\rightarrow B_1$	$m_4 \rightarrow B_2$		
517		verify m_4 signature of m_1 using V		
518 519		if verified, erase m_1 event		
520			verify m_4 signature of m_1 using V	
521 522			if verified, erase m_1 event	

Fig. 5

500